Random

Objective

In this projectlet, we venture into random numbers. In particular we will explore the ‘quality’ of random number generators available. In most explorations of this kind, a great way to begin is to visualize using graphs/plots. This will be our approach.

Specifications

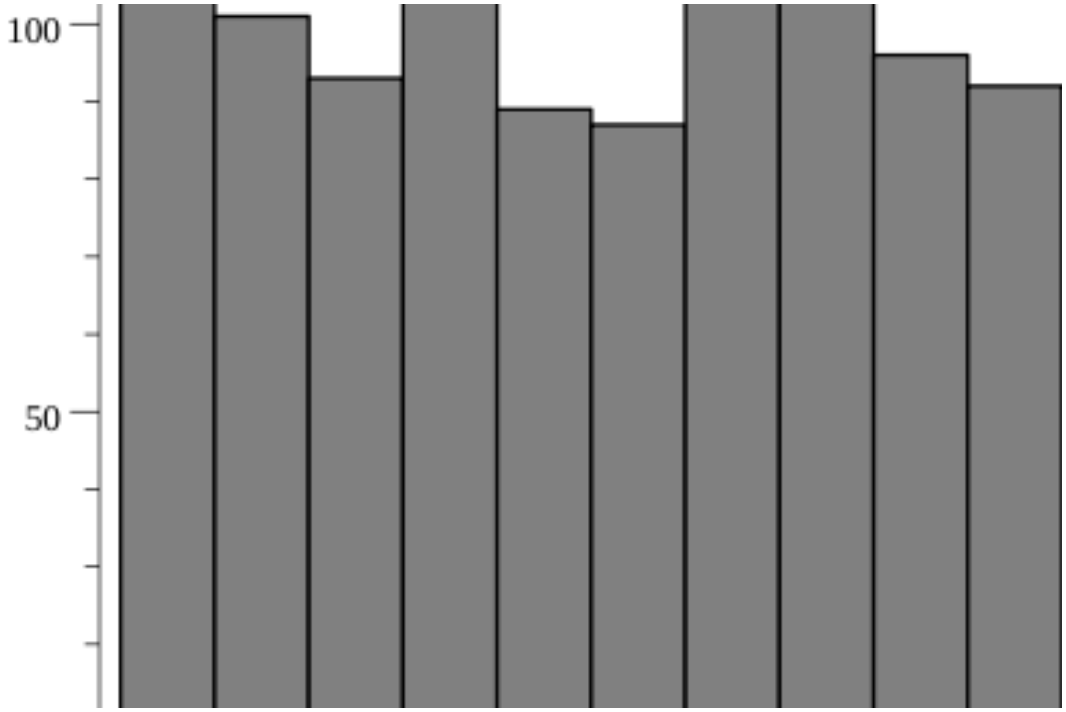
Command	Switch	Description
uniform		Uniform random numbers
		Arguments are: Minimum maximum Default: 0.0 1.0
normal		Normally distributed random numbers
		Arguments are: Mean Standard Deviation Default: 0.0 1.0
	—output	Output file name. Creates a png file.
	—series	Output plot is a series. By default it will be a histogram (configure with —slices)
	—samples	Count of number of samples. Applicable to all distributions. Default 1000
	—slices	Number of bins for histograms
	—seed	Seed the series with this number. Default 1729
	—table	Generates a table of the numbers generated - in addition to the plot. Value is the filename.

Example usage

bin/app --help

```
      This utility generates random numbers and plots the data as a series
or as a histogram.

Usage:
  random [command]

Available Commands:
  help        Help about any command
  normal      Normal distribution
  uniform     Uniformly distributed random variables
  version     Report the version of the application

Flags:
  -h, --help            help for random
  -o, --output string   output file name (default "plot.png")
  -s, --samples int     number of samples (default 1000)
  -d, --seed int        seed for random numbers (default 1729)
      --series          generate time series plots. default is histogram
  -c, --slices int      number of slices - for histograms (default 10)
  -t, --table string    tabular output filename
      --verbose         be verbose
  -v, --version         version for random

Use "random [command] --help" for more information about a command.
```

Uniform Distribution Examples



UNIFORMLY DISTRIBUTED 1000 RANDOM NUMBERS - HISTOGRAM

Implementation Examples

Go language example

https://gitlab.com/RajaSrinivasan/random.git